

計算と証明のあいだで

著者	外山 芳人
雑誌名	東北大学電通談話会記録
巻	87
号	2
ページ	41-48
発行年	2019-03
URL	http://hdl.handle.net/10097/00126501

最終講義

計算と証明のあいだで

Between Computation and Proof

外山 芳人*

Yoshihito TOYAMA



本日の最終講義では、なぜ私がこのような研究人生を歩むことになったかを話してみようと思います。私は、1952年に新潟県長岡市で生まれました。小学校時代はどんな時代かといわれていますと、日本の高度成長が始まった昭和30年代、最近公開された「三丁目の夕日」という映画で描かれている世界です。そのころは、ブラウン管の白黒テレビがまだ珍しく、近所のテレビがあるお宅に、夕方みんなで見に行くような時代でした。

私の研究の原点は、小学校のときに近所の友達のお宅にテレビを見に行ったときの出来事です。テレビでは「20世紀」というタイトルのアメリカのドキュメンタリー番組をやっていたのですが、たまたまその日は人工知能の特集でした。磁気テープのリールがぐるぐる回転する巨大なコンピュータなどが紹介されたりしたんですが、その中で印象に残ったのがウォルターのカメとシャノンのネズミでした。

ウォルターのカメというのは、サイバネティックス関係の方々だと思われませんが、ウォルターというイギリスのエンジニアが作ったバケツぐらいの大きさのロボットです。その当時は集積回路どころかトランジスタもなかったので、真空管と光電管が使われており、光の方向にカメのようにのそのそ動きます。カメの頭には小さなランプがついているので、鏡に反射した自分の姿に向かって進んだりして、単純なメカニズムにもかかわらず生き物のような複雑な動作をします。

それから、シャノンのネズミというのは、シャ

ノンという情報理論で有名なアメリカの数学者が作った、迷路の中を通り抜ける小さなおもちゃのネズミです。迷路の中にネズミを置くと、何回か失敗した後で、最後はうまく迷路を通り抜けます。鉄製のネズミを動かすのは迷路の下に磁石で、それが動きながらネズミを動かします。当時はマイクロコンピュータなどはないので、スイッチング回路の組み合わせで迷路を抜け出す方法を発見するわけです。ウォルターのカメやシャノンのネズミの映像を見て、その原理に非常に興味を惹かれました。

中学校時代の大きな出来事は、本屋でサム・ロイドやマーチン・ガードナーなどのアメリカの数学パズルの大家の本を発見して、数学パズルの面白さに目覚めたことでした。このときの数学パズル好きはその後ずっと続いていて、後には自作の数学パズルを雑誌に発表したりすることになりました。

高校時代になると、シャノンのネズミのような知的動作をする機械を自分で作ってみたいと思うようになりました。しかし、その当時は集積回路などではなくて、論理回路を作るためには、リレーというスイッチング素子やトランジスタなどを多数組み合わせる以外に方法はなく、お金のない高校生には手の届かないものでした。しかし、論理回路の設計だけなら本屋で買ってきた専門書で勉強できるので、いろいろな論理回路を紙に書いて、その動作を想像して楽しみました。論理回路の設計というのはまさにパズルで、はまり込むと非常に面白いものでした。

* 東北大学 名誉教授

私の初めてのプログラムは、デジコンというプラスチック製のコンピュータ上で動く石取りゲームのプログラムで、高校3年生のときに作りました。デジコンは米国製の教育玩具で、日本では1970年ごろにABC商会という輸入代理店から丸善などの書店経由で販売されました。

私の住んでいた長岡市のような田舎の小都市の本屋にも、デジコンが一つ展示されていましたが、外箱が透明なビニールで梱包されていて中身を見ることができない。箱に書かれている英語の説明を読むとどうも本物のコンピュータらしい。高校生の小遣い2カ月分ぐらいの値段で高かったのですが、さんざん迷った末に思い切って買い求めました。箱を開けてみると、予想通り自分で組み立てるプラスチック製のデジタルコンピュータでした。コンピュータの動作は、クロック板というプラスチックの板を左右に手でガシャンとスライドさせると1サイクルになるという非常にゆっくりした手動式です。

デジコンのプログラムは、いくつかの突起の部分にプラスチックの筒をはめ込むことで行い、クロック板を左右にスライドさせることによって筒をはめ込んだ突起が押されて、論理回路の動作を実現します。出力はプラスチック板をスライドさせて小さな窓に0と1を表示する。デジコンの簡単なプログラム集がついていて、二進数の四則演算や、簡単なパズルを解くプログラムが載っていました。さっそく、自分でもいろいろなプログラムを作ってみました。

自分で考えたプログラムのひとつは、石取りゲームというパズルを解くプログラムでした。そのプログラムをABC商会に送りました。今回の最終講義の準備で古文書を探していたら、ABC商会からの返事の手紙が残ってありました。手紙には、あなたのプログラムは大変面白いので、デジコンのプログラム集の第2集を出版するときにはぜひ掲載したい。ついては、お礼の2,000円を同封する、という内容が書かれていました。これは私が初めて手にした原稿料でした。ただ、デジコンに夢中になる子供はあまりいなかったようで、楽しみにしていた第2集は残念ながら出版されませんでした。

長岡市の実家から歩いて10分ぐらいの新潟大

学工学部に入学したのは1971年でした。興味があったのは情報科学でしたが、当時の日本には情報科学関係の学科はほとんどなかったので、コンピュータが勉強できそうな電子工学を専攻しました。しかし、電気回路、電磁気学、電子回路、量子力学などの伝統的な電子工学の授業ばかりで、コンピュータ関係の授業はほとんどない。せいぜい論理回路の授業があるくらいです。また、電子工学科のコンピュータもHITAC-10というミニコンが1台あるだけで、学部学生は触ることもできません。

大学の授業には興味がわからないので、コンピュータ関係の専門書を本屋でいろいろあさっているうちに、コンピュータの基礎理論であるオートマトンや言語理論、計算の理論などの分野を発見しました。これらは、コンピュータの実物がなくても紙と鉛筆だけで楽しめ、しかも数学パズルのような面白さをもっています。万能チューリングマシンを知ったときには、紙の上でのプログラム作りに夢中になりました。数学パズルはこれまで遊びだと思っていたのですが、パズルがそのまま学問になるような面白い分野が世の中にあることを知ったのは、私にとって大きな驚きでした。

そのようなわけで、学部時代は電子工学はそっこのけで、コンピュータの基礎理論の専門書を本屋でいろいろ買い込んで読みふけりました。私が大学に入学した翌年に出版された本多波雄先生の「オートマトン・言語理論」という本では、演習問題を全部解いて基礎を身につけました。また、ミンスキーの「計算機の数学的理論」という数学パズルの宝石箱みたいな本では、神経回路網とオートマトンの関係、セル・オートマトンの一斉射撃問題、最小万能チューリングマシンの作り方などの面白い話題を楽しむことができました。

せっかく夢中になれる学問分野を発見したのですが、新潟大学では指導してくださる先生はおりません。そこで、「オートマトン・言語理論」の著者の本多波雄先生に弟子入りしようと考え、大学院への進学は東北大学の情報工学専攻を選びました。ところが、本多先生は1年後に名古屋大学に移られることが決まっていたので、本多研究室に学生の配属はなく、残念ながら弟子入りの希望はかないませんでした。結局、本多先生のお弟子さんの木村正行先生の研究分野が似ていたので、

木村研究室に入れていただくことになりました。

木村研究室は、木村先生のお人柄もあると思うのですが、非常に自由な研究室で、大学院の学生を一人前の研究者として尊重する一昔前の大学の雰囲気を残していました。「君たちは、もう自分でできるはずだから、自分で考えてやりなさい」ということで、ある意味では放任主義なのかもしれませんが、本当に好きなことを自由に研究させていただけました。また、研究室のゼミや輪講では、助教授になられて間もない丸岡章先生や、助手になられたばかりの阿曾弘具先生に、ずいぶん鍛えていただきました。

研究室のゼミで勉強したのは、出版されたばかりの Aho、Hopcroft、Ullman の「Design and Analysis of Computer Algorithms」という本で、これは今日では情報科学分野の古典となっています。ゼミでは、難しい演習問題に挑戦するのが面白くて、夢中になって取り組みました。このゼミで一番勉強になったのは、帰納的な構造の取り扱いでした。帰納法で何かを証明しようと思ったら、対象も帰納的な構造をしていなければいけない。だから、構造が帰納的であることと、帰納法で証明することは常に対になっているということを、多くの演習問題を解くことで自然に身につけることができました。

研究室では、学生が集まっていろいろな本を勉強する自主ゼミがいくつかありまして、バーコフ・マクレーンの「現代代数学概論」、この本は代数学の有名な古典ですが、その自主ゼミに参加させていただきました。ドクターの先輩たちと一緒にこの数学書を読んだことは、私の人生で得難い経験でした。

コンピュータの基礎理論を勉強していたときには、オートマトンとかチューリングマシンなどの抽象的な概念が、生き生きとした具体的なイメージとして理解できました。ところが、抽象的な代数ではそのような感覚をもつことがどうしてもできないのです。本の中の証明を機械的に追うことはできるのに、生き生きとしたイメージは浮かばず、本当に理解したような気がしない。しかし、ゼミに参加しているうちに、ある日突然目からうろこがとれたように、抽象的な世界のイメージが心に浮かぶようになり、数学の本を読むというの

はこういうことなのかと気づきました。このゼミで数学の本の読み方のコツがつかめたのは、私にとって一生の財産でした。

木村研究室では、修士研究のテーマは学生が自分で決めることになっていて、よほどのことがないかぎり先生や先輩がこのテーマをやりなさいとは言いません。このような自由な環境はとてもよかったです。研究テーマを自分で決めるためには、その研究分野の最前線の状況をまず知らなくてはいけない。その部分も含めて自分ですべてやりなさいということです。そもそも、研究分野の最前線まで到達できたら、修士研究の 8 割から 9 割はすでに終わっているようなものです。そういう意味では、学生にとってずいぶん厳しい研究室だったと今ではわかるのですが、若いときにはそんなことには気がつかないものです。

私の修士研究ですが、その当時興味をもっていたのは学習の話で、生物の学習能力を機械で実現する方法を研究してみたいと思いました。ネズミの学習能力を調べるために、ネズミを T 字路で走らせ、一方の側ではチーズを与え、反対側では電気刺激を与える実験があります。例えば、ランダムに 7 割ぐらいは右側にチーズを置き、3 割ぐらいは反対側にチーズを置くと、実験を繰り返すうちにチーズをもらえる確率の高い右側をネズミは選ぶようになります。

このような学習能力を解析するためのモデルとして、学習オートマトンというものが当時提案されていました。しかし、その振る舞いを調べるためにはもっぱら計算機シミュレーションが用いられており、理論的な解析はほとんど手付かずでした。そこで、修士研究で学習オートマトンの振る舞いを理論的に明らかにしてやろうと決心しました。

ところで、学習オートマトンの解析は、実際にはオートマトンの解析というよりは、連続状態の吸収マルコフ過程という確率過程の解析です。オートマトンの状態が有限でしたらマルコフ連鎖なので解析は容易なのですが、状態が連続無限の確率過程なので解析は非常に難しい。私は電子工学科出身ですから、このような確率過程を解析するための基礎知識がありません。仕方がないので、先輩達にいろいろ教えてもらって、まずルベーグ

積分の薄い入門書の勉強から始め、それから確率過程論の入門書を勉強しました。その後は、手当たりしだいに確率過程や確率制御関係の本をいろいろ調べ、確率的リアプノフ関数をうまく使えば問題が解けるかも知れないことに気づきました。

そこで、学習オートマトンの振る舞いを確率的リアプノフ関数で解析することに取り組んだのですが、解析の鍵となる不等式が証明できません。毎日一生懸命考えたのですが、どうしても証明することができない。そもそも誰かに与えられた問題ではなく、自分で勝手につくった問題なので、本当に答えがあるかどうかともわかりません。修士2年の夏休みになっても、問題解決のめどがまったくたらず、頭も空回りするばかりで、精神的にずいぶん追い詰められました。

夏休みの帰省中に実家に帰って近所をぼんやり散歩していたときのことです。これまでいくら考えてもわからなかった不等式の証明が突然ひらめました。最初は信じられなくて、頭の中で何回もチェックしましたが確かにうまく証明されている。これまで誰も解けなかった問題が解けた瞬間の高揚感は、今でも鮮明に覚えています。これが、研究を一生の仕事にしようと私が決心したきっかけです。

修士課程修了後、電電公社（現 NTT）に就職しました。なぜ電電公社に就職したのかというと、ある日、電電公社の武蔵野研究所の池野信一先生が東北大学で講演を行いました。講演タイトルは回路網理論に関するものだったように思いますが確かではありません。というのは、池野先生は講演タイトルとは無関係な、その当時珍しかったマイクロプロセッサを使った自作のコンピュータについて話されてからです。池野先生のコンピュータはすべて手づくりで、例えば、基板を差すラックさえも木工工作で、のこぎりでつけた筋目に基板を差し込むように工夫されています。自作の苦勞などを楽しそうに話される池野先生にすっかり引きつけられて、電電公社の研究所に就職することに決めました。

電電公社では、計算機関係の研究をしている基礎第1研究室に配属されました。池野先生の特別研究室には新入社員は配属されず、本多先生のときと同様、今回も弟子入りはかないませんでした。

基礎第1研究室は、コンピュータのアーキテクチャやソフトウェアなどの基礎研究を行う研究室で、私が入った当時は大規模な並列計算機の開発プロジェクトが進んでいました。

私の初めての仕事は、研究室の次期プロジェクトのための調査でした。その当時、データフロー計算機という新しい原理の計算機が世の中で話題になっていて、次のプロジェクトの候補のひとつでした。そこで、新入社員の私が調査を命じられ、2年くらいかけてデータフロー計算機の調査資料をまとめました。その後、研究室ではデータフロー計算機の開発プロジェクトがスタートしましたが、私はもっと理論的なことをやりたいという気持ちが強くて、このプロジェクトには参加しませんでした。

John Backus という方が、1977年にチューリング賞というアメリカの計算学会の有名な賞をとられました。このときの彼の記念講演が、世界中のコンピュータ研究者に大きな衝撃を与えました。Backus は IBM の研究者で、世界最初の高級プログラミング言語 FORTRAN の開発プロジェクトのリーダーとして有名でした。チューリング賞は、その業績に対して与えられたわけです。ところが、彼はその記念講演で、彼の開発した FORTRAN のようなプログラミング言語はだめだ、もっと別の言語を考えるべきだと主張したのです。これにはみんな驚きました。

なぜだめかということ、現在のコンピュータというのは、フォン・ノイマン型コンピュータであり構造上の大きな欠点がある。多くのプログラミング言語はこのコンピュータ上で効率よく動作するように設計されているため、コンピュータの欠点がそのまま言語に反映されてしまう。そのような言語をもちいて大規模なプログラムを開発しようとするから、解決困難なさまざまなソフトウェア上の問題が出てくる。だから、既存のプログラミング言語やプログラムなどを改良するのではなく、諸悪の根源であるフォン・ノイマン型コンピュータそのものを捨て去り、まったく異なる原理の計算モデルから出発すべきだ。それによって、関数的プログラミング・スタイルと名付けられた新しい発想のプログラムが可能になるはずだ。このようなことを Backus は主張したのです。

Backus のこの講演が、ここにいらっしやる大堀先生や住井先生が現在研究されている関数型プログラミング言語の出発点です。Backus の講演がきっかけとなって、従来のフォン・ノイマン型言語とはまったく異なる発想の言語を開発しようという機運が世界中で一気に高まったのです。

私も Backus の講演に非常に感銘を受け、新しい原理の計算モデルをぜひ研究したいと、周りの人たちや上司を一生懸命に説得しました。上司も最後はあきらめて、1 人ぐらいはデータフロー計算機の開発プロジェクトから外れてもよいだろうということで、ある程度自由に研究をやらせていただけることになりました。

そこで、フォン・ノイマン型計算モデルとは異なる新しい計算モデルは何かということが問題となります。Backus 自身もフォン・ノイマン型計算モデルはだめだとは言っているが、どのような計算モデルならよいのかという具体的な答えまでは与えていません。ただ、彼は計算モデルの必要条件として、エレガントで簡潔な数学的記述で表現できること、それから、計算モデルの重要な振る舞いが理論的にきちんと解析できることをあげています。

それでは、そのような条件をみたす計算モデルは何だろうか。誰もそのようなモデルは見つけていないので、自分で考えるしかありません。さまざまな計算モデルを調べてみましたが、どれも私にはぴんとこない。ただ、Backus は、論理的な基礎としてラムダ計算とかコンビネータなどをあげていたので、まずこれらを勉強してみようと考えました。

修士研究のときに、ルベーク積分から勉強したのと同じようなもので、まず基礎から手を付けたわけです。ただ、この当時はラムダ計算の適当な教科書がなく、Henk Barendregt 先生のラムダ計算の有名な教科書もまだ出版されていなかったもので、勉強するのは苦労しました。ラムダ計算について勉強していくうちに、計算順序を一意に定めない非決定的な計算モデルであるラムダ計算では、計算過程の合流性が論理構造と計算構造を結びつける鍵となっていることに気づきました。

ここで、計算過程の合流性とは何かということ

について簡単に説明したいと思います。等式 $1 + 2 = ?$ の右辺は何ですかと聞かれたら、多くの方は迷うことなく $1 + 2 = 3$ と答えるでしょう。しかし、 $1 + 2 = 10 \times 10 - 97$ とか $1 + 2 = 1 + 2$ と答えても、等式の左辺と右辺は等しいので、論理的には間違いではないはずです。なぜ、私たちは $1 + 2$ の答えを 3 と考えるのでしょうか。

等式そのものにはもともと計算という意味はありません。たとえば、等式 $1 + 2 = 3$ は左辺と右辺が等しいという論理的な意味をもつだけです。したがって、論理の世界では $1 + 2$ から 3 を得ることもできますが、逆に 3 から $1 + 2$ を得ることもできる、つまり、等式は、左から右にも行けるし、右から左にも行ける、平面的な世界を表しています。

一方、計算の世界では $1 + 2$ の計算結果として 3 が得られるのであり、3 の計算結果として $1 + 2$ が得られることはありません。つまり、計算の世界は論理の世界とは異なり、高低差のある世界なのです。そこで、等式 $1 + 2 = 3$ を複雑な式から単純な式への非可逆な書き換え規則 $1 + 2 \rightarrow 3$ とみなすことで、等式で表された平面的な論理の世界を、高低差のある計算の世界に自然な形で結び付けることが可能となります。

高低差のある計算の世界では、書き換え規則を適用することによって高い地点から低い地点に進んで行くことが計算であると考え、これをリダクションとよびます。複雑な式がリダクションによってどんどん簡単な式となり、最後は答えに行き着くわけです。

ところで、リダクションの道筋は一通りとはかぎりません。同じ出発点から出発しても、どのように書き換え規則を適用するかによって何通りものリダクションが可能となるからです。このとき、どのようにリダクションしても答えが一意に定まるなら、この計算の世界は合流性をもっているといえます。たとえば、複雑な四則演算の式の計算では、どのような順番で計算を進めても常に同じ答えが得られます。したがって、四則演算の世界は合流性をもっていることになります。

ラムダ計算では計算過程の合流性が本質的であ

ることに気づいたので、ラムダ計算以外の非決定的な計算モデルについても合流条件を調べてみようと考えました。非決定的な計算モデルとして取り上げたのは、そのころ研究が開始されたばかりの項書き換えシステムでした。

項書き換えシステムは、等式を計算規則とみなす計算モデルです。これが、Backus の要求した2つの条件をみたしているかどうか考えてみましょう。等式というのは小学生でも理解できる単純な論理構造です。ですから、エレガントで簡潔な数学的記述で表現できるかという条件については合格です。次の、重要な振る舞いがきちんと解析できるかという条件については、合流条件を理論的に明らかにできれば合格するはずです。

当時知られていた項書き換えシステムの合流条件は、左線形のシステムか、あるいは、停止するシステムに限られていました。しかし、停止せずに左線形でもないシステムの合流条件については、解析が難しいため、まったく手がついていません。そこで、非線形で停止しないシステムの合流条件に焦点を合わせて研究することにしました。しかし、いくら考えても解析方法が思いつきません。

あるとき、海外の学会誌に掲載されている博士論文の要約集を調べていると、私と同じような興味をもつオランダの研究者を発見しました。その後、ずいぶんお世話になったオランダ国立数学・計算機科学研究所の Jan Willem Klop 先生です。さっそく手紙を書いて、博士論文を送っていただきました。Klop 先生の博士論文には、非線形で停止しないにもかかわらず合流性をもつ書き換えシステムがいくつか示されていました。しかも、これらの合流性はシステムを少し変形するだけで失われてしまいます。

ほとんど同じように見える二つのシステムで、一方は合流性を持ち、他方は合流性をもたない理由は何かを考え続けました。やがて、合流性をもつシステムでは直和分解できるのに、合流性をもたないシステムでは直和分解ができないことに気がつきました。そこで、このようなモジュラ性はどのような項書き換えシステムに対しても成立する一般的な性質ではないかと予想して、その証明に取りかかりました。

合流性がモジュラであるという予想は、これまで類似研究がまったくなかったので、成立するかどうかは五分五分でした。なかなか証明は進まなかったのですが、このように数学的に美しい定理が成立しないはずはないと自分を励ましながら、1年くらいかけてようやく証明に成功しました。これが、外山の定理と現在よばれているものです。

この研究でとくに苦勞したのは、これまで誰も考えつかなかった枠組みなので、私の頭の中ではすでに自明となっている証明を、他人が理解できるように形に記述することでした。そのため、いくつかの新しい概念を導入し、それらを簡明に表現できる記法を工夫しました。苦勞のかがあって、これらの記法の多くは、今日でも書き換えシステムの研究で広く使われています。

項書き換えシステムの合流性はモジュラであるという結果を論文にまとめて、海外の論文誌に投稿しました。すると、合流性だけではなく、停止性がモジュラであることも同様な手法で容易に証明できるはずなので、その結果も合わせて論文に入れたらどうかとのコメントを、レフリーからもらいました。このレフリーが Klop 先生であることは後からわかったのですが、Klop 先生だけではなく、この分野の研究者たちはみんな、合流性がモジュラなのだから停止性もモジュラであることは当然であると予想したのでした。

私は、論文を投稿したときに、停止性がモジュラであることをすでに示していました。私の研究ノートには、前半に合流性のモジュラの証明のスケッチ、後半に停止性のモジュラの証明のスケッチが書かれています。論文誌に投稿した論文は、前半部分のスケッチをまとめたものでした。そこで、レフリーのコメントにしたがって、後半部分のスケッチの完全な証明をつくり始めました。ところが、スケッチの中で仮定している性質を証明しようすると、別の仮定がいくつか必要となり、この繰り返しでいつまでたってもゴールに到達できません。絶対証明できるはずだと信じて1年くらい頑張ったのですが、どうしても証明を完成させることができず、すっかり疲れ果ててしまいました。

ある朝、通勤の途中の横断歩道で信号待ちをしていました。信号が青に変わって歩き始めた途端、

反例があるかも知れないという考えが突然に頭に浮かびました。そして、横断歩道を渡り終ったときには、単純な項書き換えシステムが頭の中に自然に現れていました。みんなの予想に反して、停止性がモジュラではないという反例でした。これは、外山の反例として今日でも広く引用されています。

これによって、合流性はモジュラであるという外山の定理と、停止性はモジュラでないという外山の反例を、並べて論文にすることができました。このモジュラ性という概念は、その後多くの論文で引用されるとともに、教科書等でも広く紹介され、この分野の研究に大きな影響を与えたと思います。

外山の定理や反例がきっかけで、オランダの国立数学・計算機科学研究所に 1 年間滞在させていただくこととなり、1990 年夏に妻と 2 才の長男を連れてアムステルダムに渡りました。オランダでは、Klop 先生や Barendregt 先生の研究グループとの共同研究を行いました。同じような興味をもつ人たちが集まってさまざまなテーマで活発に議論したり、外国の一流の研究者たちが絶えず訪ねてきて意見交換したりするコミュニティの存在はとても印象的でした。また、研究以外にも、オランダで知り合った多くの方々と家族を含めた交流によって、実り多い 1 年間で過ごすことができました。

この当時、項書き換えシステムは生まれたばかりの研究分野で、日本ではほとんど知られておりませんでした。そこで、日本にもオランダのような研究コミュニティがぜひ欲しいと思い、オランダから帰国した 1991 年に、三重大大学の大山口通夫先生と相談して、TRS ミーティングという研究会を立ち上げました。

研究会でのルールは 2 つ決めました。1 つめは、参加者は必ず何か発表すること。項書き換えシステム以外のテーマでも、面白いと思えば何を話してもよいから、必ず何か発表すること。2 つめは、発表は英語で行うこと。海外の研究者も参加するので、とにかく英語で発表することを決めました。

第 1 回 TRS ミーティングを開催したのは 1991 年 12 月です。仙台で先月第 48 回を開催しましたの

で、年 2 回くらいのペースで 30 年近く続いたことになります。こんなに長く続いたのは、ルールがゆるやかで、この分野に興味をもたれた方でしたら誰でも歓迎したからだと思います。また、発表時間も柔軟で、英語が不得意な学生の方も参加しやすいように、10 分でも 15 分でもよいから何か話してもらおうようにしました。また、海外の研究者が日本に滞在する場合や、国際会議などで多くの研究者が日本を訪問する場合には、彼らが参加しやすい時期に開催しました。

TRS ミーティングには、海外の研究者が毎回参加するので、世界の最前線の研究状況が、国内の若手研究者に直接伝わるようになりました。また、国内の若手研究者の研究成果を海外の研究者に直接発信する場としても非常に有効でした。30 年近く続けたおかげで、はじめは学生として参加されていた方々が、今では先生として自分の学生を連れてくるようになり、この分野のコミュニティを日本に根付かせることができたと思います。

オランダから帰国すると、恩師の木村先生から設立したばかりの北陸先端科学技術大学院大学 (JAIST) に来ないかとお誘いがあり、1993 年に NTT を退職して教授として着任することになりました。研究室の助教授は、酒井正彦先生が名古屋大学からいらっしゃいました。

JAIST は、交通手段もほとんど整備されていない山の中に開学したばかりで、1 年ごとに新しい建物を 1 棟ずつ増やししながら、教育・研究環境を少しずつ整備していくような状況でした。教員も学生も苦勞が多かったのですが、何もないところから新しい大学を一緒に作っていく楽しさを、学生の人たちと味わうことができました。

JAIST で 7 年間過ごした後、縁があって 2000 年 4 月に東北大学の電気通信研究所に教授として着任しました。東北大学では、計算システムと証明システムの融合を目指して、書き換えシステムに基づくプログラムの自動検証や自動変換、定理自動証明などの研究に取り組みました。これらの研究では、私の得意な理論的なアプローチだけではなく、学生やスタッフの協力によって実験的なアプローチもいろいろ試みることができました。

実験的アプローチによる成果のひとつは、世界最初の合流性自動証明システム ACP の開発です。これは、当時の研究室の准教授で、現在は新潟大学の教授をされている青戸等人先生との共同プロジェクトで、2007年から開発を始めました。ACP は与えられた書き換えシステムが合流性をもつかどうかを、モジュラ性に基づく分割統治法によって自動判定します。

合流性自動証明システム ACP を2009年の国際会議で発表すると、これが契機となっていくつかの研究グループで合流性自動証明システムの開発が始まりました。さらに、自動証明システムの国際的競技会も2012年から毎年開催されるようになりました。この競技会からは、合流性自動証明のためのさまざまな新しいアイデアが毎年生まれています。

最後に、なぜ私は研究するのだろうかということについて、少し話してみたいと思います。基礎的な理論研究者の間では、発表者を立ち往生させたければ「その研究は何の役に立つのですか」と質問すればよい、というジョークがあります。つまり、面白そうだが実際には何の役にも立たない無意味な研究ではないか、という意地悪な質問です。

私は企業で16年間、大学で25年間研究生活をおくりました。とくに、企業の研究所では、その研究が何の役に立つのかということを大変重視します。ですから、役立ちそうもない基礎研究をしていると、この研究は本当にやる意味があるのだろうかかと悩むこともあります。そのようなときには、ファインマンという有名な物理学者の以下の言葉を思い出すことにしています。

「前にはあんなに物理をやるのが楽しかったというのに、今はいささか食傷気味だ。なぜ昔は楽しめたのだろう。そうだ、以前は私は物理で遊んだのだった。いつもやりたいと思ったことをやったまでで、それが核物理の発展のために重要であろうがなかろうが、そんなことは知ったことではなかった。ただ私が面白く遊べるかどうかが決めた手だったのだ」。

私も、この研究が何の役に立つのかという迷いが出てきたときには、何の役にも立ちもしないけ

れど、面白いからやっているのだと考えることにしています。研究が役に立つかどうかは、研究者本人が決めることではありません。それは利用する人が決めることで、本人が役に立つといくら主張しても本当はまったく役に立たないこともあるし、本人がこんなものは役に立たない、面白いからやっているだけだと言っても、本人が予想もしなかった形で役に立つこともあります。ですから、研究で大切なことは役立つかどうかではなくて、その人でなければできない独創性のある研究であること、何年たっても風化しない深度の深い研究であること、それが本当に大切なことだと思います。そして、そのような研究は、研究者本人が面白くて夢中になってこそ生まれるものなのです。

最後に、面白いことを好き勝手に楽しんできた私の研究人生を、いろいろな形で今日まで支えてくださった皆様に心から感謝いたします。